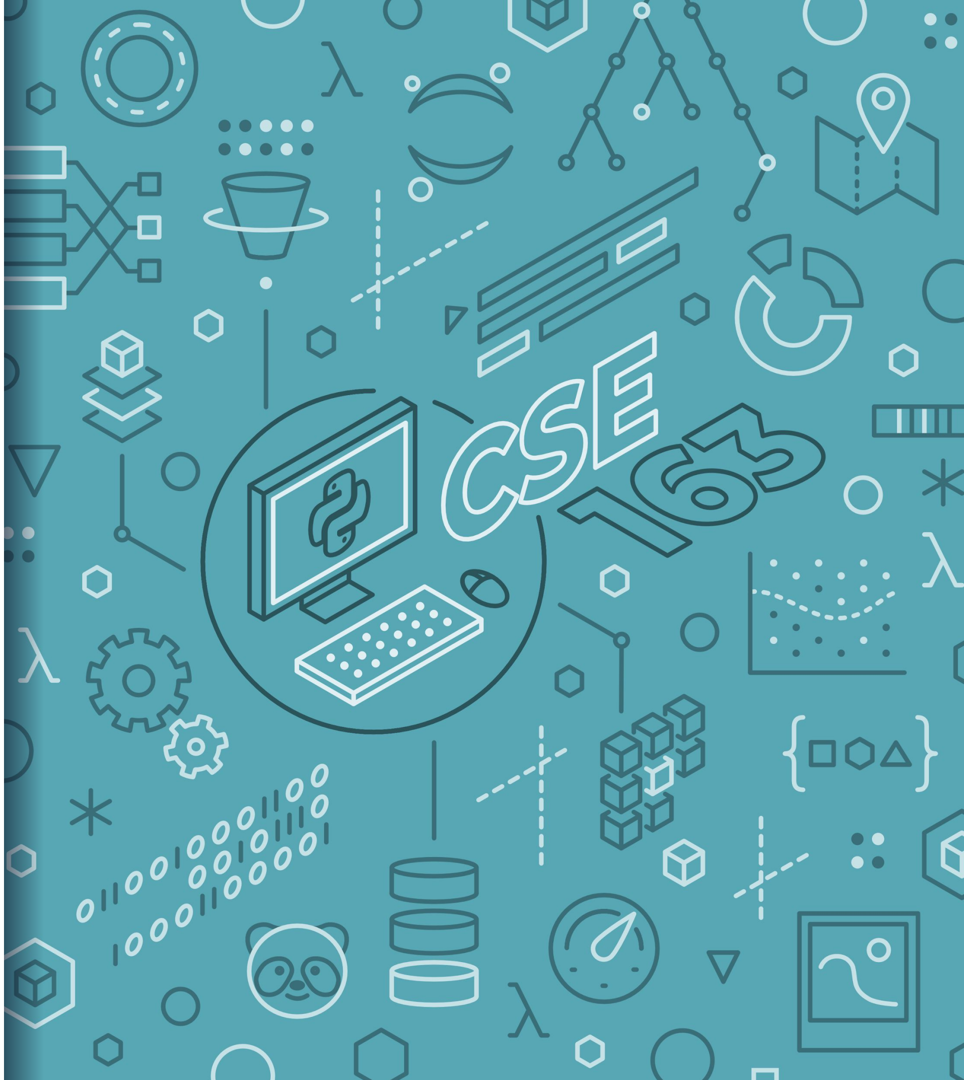
CSE 163

Images + Machine Learning

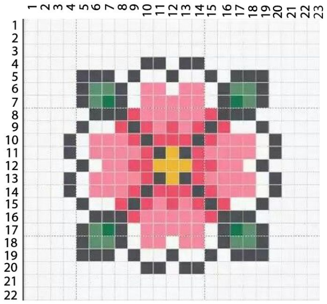
Hunter Schafer

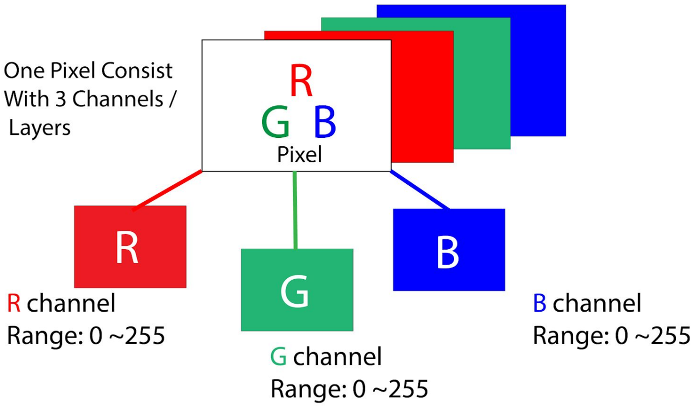


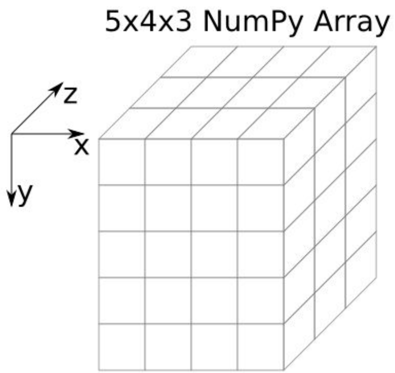
Color Images

Each pixel coordinate (x, y) contains 3 values ranging for intensities of 0 to 255 (8-bit)
- Red - Green - Blue

Mixing different intensities of each color gives us the full color spectrum.







Convolution

When wanting to use "local" information, we need to use a sliding window approach (i.e. a **convolution**)

Move the sliding window across the image, and compute the sum of the element wise product of the window (kernel) and image

Image

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

Kernel

0	1	2
2	2	0
0	1	2

Poll Everywhere

Think

1 minute

pollev.com/cse163

What is the result of applying a convolution using this kernel on this image?

Image

1	2	3
4	5	6
7	8	9

Kernel

1	2

Poll Everywhere

Pair

1.5 minutes

pollev.com/cse163

What is the result of applying a convolution using this kernel on this image?

Image

1	2	3
4	5	6
7	8	9

Kernel

1	2

Machine Learning

Remember back to the land of machine learning

We have some data which we map to **features** and **labels**

We use a machine learning **algorithm** to train a **model** on that data

One major goal of machine learning is to produce a model that is accurate on future data.

- This is why we need that test set to help us validate our model

ML + Images

How do we do machine learning on images?

- **Simplest: Unroll the image into a vector**
- Complex: Use other tools to extract features from the images

Raw Image

10	20	30
40	50	60
70	80	90

Unrolled Image

10
20
30
40
50
60
70
80
90

ML + Images

Pros: Simple transformation (just a call to `reshape`!)

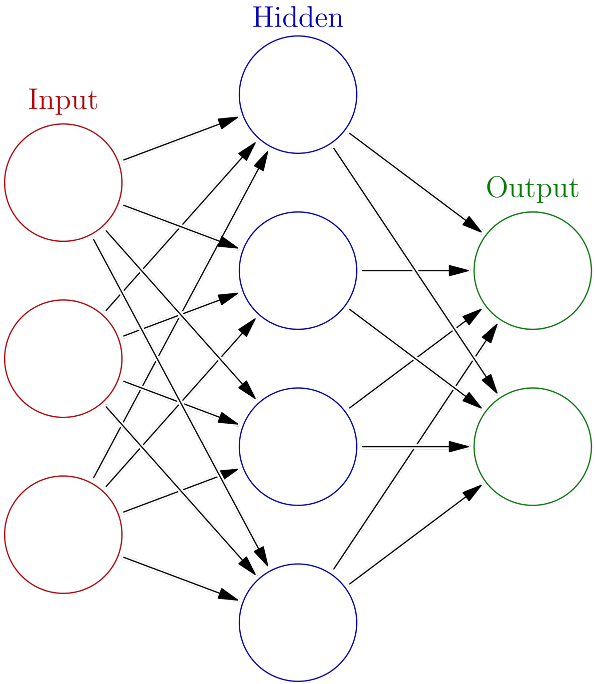
Cons: It loses the idea of “neighboring” pixels (up/down)

- Most machine learning models don’t take position of the features into account
- This is where more complex models like convolutional neural networks come in to encode that local information as features

Despite these drawbacks, it can work in practice on some problems!

Neural Network

Based on how our brains work



MNIST

Colab

MNIST is a dataset of handwritten digits and their labels

- Very popular starter project with images and ML

Lots of hyper-parameters to specify in the model!

- Learning Rate
- Number of hidden layers
- Number of hidden nodes

Unsupervised Learning

So far, we have seen **supervised** machine learning, where we have to explicitly shown the algorithm the labels

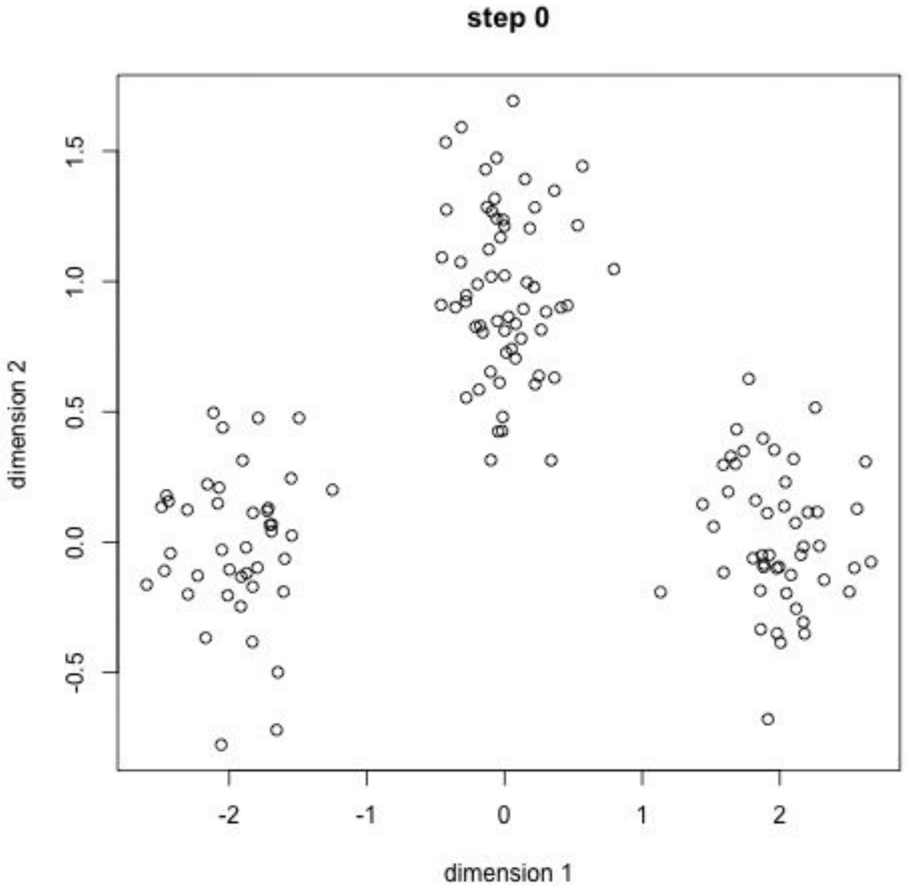
Unsupervised machine learning lets the algorithm try to learn trends on its own without providing explicit labels

Examples

- Clustering
- Outlier detection

Clustering

(k-means)



Blob Detection

Colab

- Library: scikit-image
- Blob Detection